



FOUNDATION OF SOFTWARE CONSTRUCTION

By SHOLIQ – IS Departement -
its

OUTLINE

- 1. Understanding to Software Construction**
- 2. Metaphors for a Richer Understanding of Software Development**
- 3. Measure Twice, Cut Once: Upstream Prerequisites**
- 4. Key Construction Decisions**

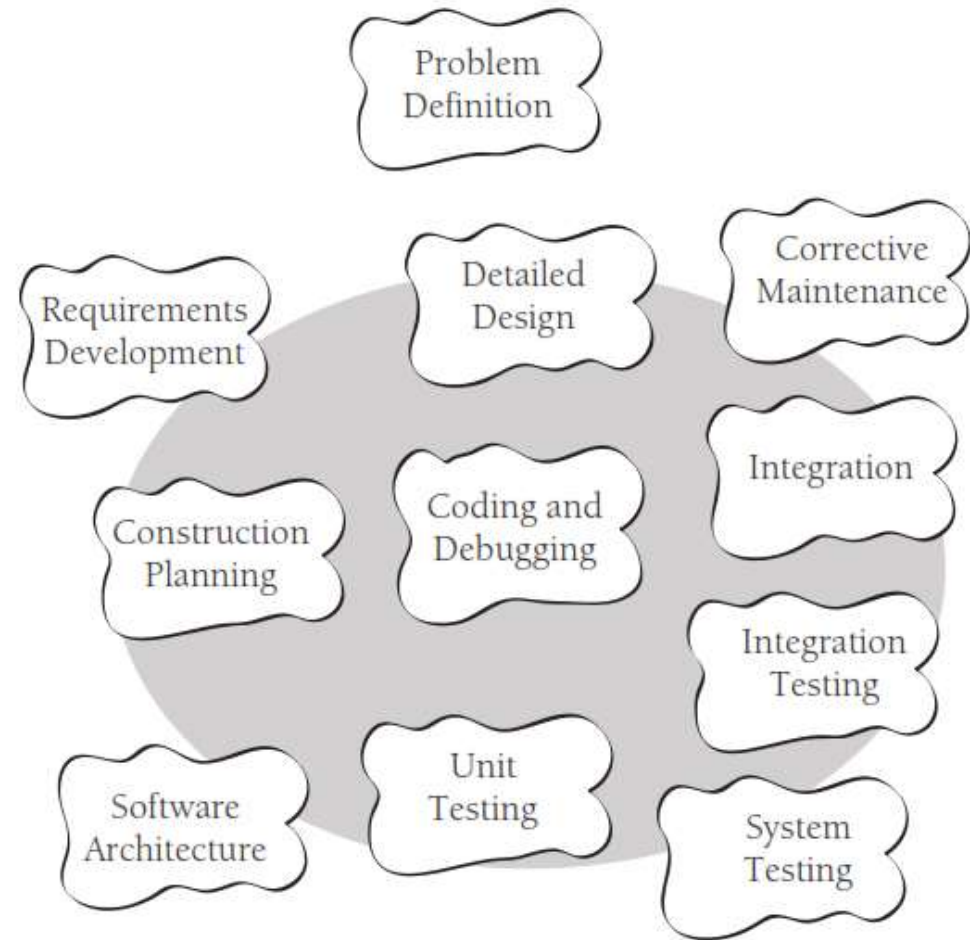
WHAT IS SOFTWARE CONSTRUCTION?

Developing computer software can be a complicated process, and in the last 25 years, researchers have identified numerous distinct activities that go into software development. They include:

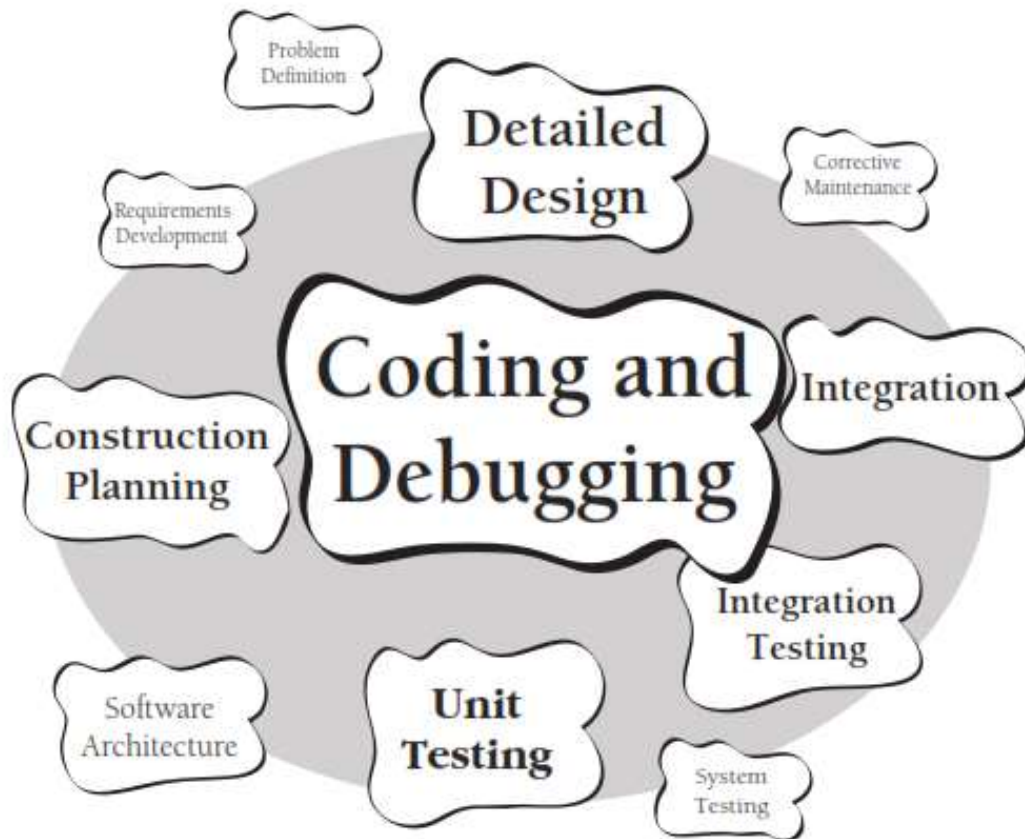
1. Problem definition
2. Requirements development
3. Construction planning
4. Software architecture, or high-level design
5. Detailed design
6. Coding and debugging
7. Unit testing
8. Integration testing
9. Integration
10. System testing
11. Corrective maintenance

WHAT IS SOFTWARE CONSTRUCTION?

Construction activities are shown inside the gray circle. Construction focuses on coding and debugging but also includes detailed design, unit testing, integration testing, and other activities.



WHAT IS SOFTWARE CONSTRUCTION?



Construction is mostly coding and debugging but also involves detailed design, construction planning, unit testing, integration, integration testing, and other activities.

Here are some of the specific tasks involved in construction:

- Verifying that the groundwork has been laid so that construction can proceed successfully
- Determining how your code will be tested
- Designing and writing classes and routines
- Creating and naming variables and named constants
- Selecting control structures and organizing blocks of statements
- Unit testing, integration testing, and debugging your own code
- Reviewing other team members' low-level designs and code and having them review yours
- Polishing code by carefully formatting and commenting it
- Integrating software components that were created separately
- Tuning code to make it faster and use fewer resources

WHY IS SOFTWARE CONSTRUCTION IMPORTANT?

You probably agree that improving software quality and developer productivity is important. Why is construction an important focus?

- ***Construction is a large part of software development*** Depending on the size of the project, construction typically takes 30 to 80 percent of the total time spent on a project. Anything that takes up that much project time is bound to affect the success of the project.
- ***Construction is the central activity in software development*** Requirements and architecture are done before construction so that you can do construction effectively. System testing (in the strict sense of independent testing) is done after construction to verify that construction has been done correctly. Construction is at the center of the software-development process.
- ***With a focus on construction, the individual programmer's productivity can improve enormously*** A classic study by Sackman, Erikson, and Grant showed that the productivity of individual programmers varied by a factor of 10 to 20 during construction (1968). Since their study, their results have been confirmed by numerous other studies (Curtis 1981, Mills 1983, Curtis et al. 1986, Card 1987, Valett and McGarry 1989, DeMarco and Lister 1999, Boehm et al. 2000). This book helps all programmers learn techniques that are already used by the best programmers.

WHY IS SOFTWARE CONSTRUCTION IMPORTANT?

- ***Construction's product, the source code, is often the only accurate description of the software*** In many projects, the only documentation available to programmers is the code itself. Requirements specifications and design documents can go out of date, but the source code is always up to date. Consequently, it's imperative that the source code be of the highest possible quality. Consistent application of techniques for source-code improvement makes the difference between a Rube Goldberg contraption and a detailed, correct, and therefore informative program. Such techniques are most effectively applied during construction.
- ***Construction is the only activity that's guaranteed to be done*** The ideal software project goes through careful requirements development and architectural design before construction begins. The ideal project undergoes comprehensive, statistically controlled system testing after construction. Imperfect, real-world projects, however, often skip requirements and design to jump into construction. They drop testing because they have too many errors to fix and they've run out of time. But no matter how rushed or poorly planned a project is, you can't drop construction; it's where the rubber meets the road. Improving construction is thus a way of improving any software-development effort, no matter how abbreviated.

KEY POINTS

- Software construction is the central activity in software development; construction is the only activity that's guaranteed to happen on every project.
- The main activities in construction are detailed design, coding, debugging, integration, and developer testing (unit testing and integration testing).
- Other common terms for construction are “coding” and “programming.”
- The quality of the construction substantially affects the quality of the software.
- In the final analysis, your understanding of how to do construction determines how good a programmer you are.